Refine Search

Search Results -

Terms	Documents
L2 and (port or channel)	37

Database:
US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search: L3

Refine Search | Recall Text | Clear | Interrupt

Search History

DATE: Wednesday, November 17, 2004 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=PGPB,USPT,USOC; PLUR=YES; OP=OR			
L3	L2 and (port or channel)	37	L3
L2	L1 and ((storage or memory or disk or disc) near5 serial$3)	40	L2
L1	((high adj1 speed) near5 bus) near10 (chipset or chip or IC or (integrated adj1 circuit))	472	L1

END OF SEARCH HISTORY

h e b b cg b e e ch

Refine Search

Search Results -

Terms	Documents
L3	0

Database:
US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search: L4

Refine Search

Recall Text Clear Interrupt

Search History

DATE: Wednesday, November 17, 2004 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
DB=EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L4	L3	0	L4
DB=PGPB,USPT,USOC; PLUR=YES; OP=OR			
L3	L2 and (port or channel)	37	L3
L2	L1 and ((storage or memory or disk or disc) near5 serial$3)	40	L2
L1	((high adj1 speed) near5 bus) near10 (chipset or chip or IC or (integrated adj1 circuit))	472	L1

END OF SEARCH HISTORY

h e b b cg b e e ch

Refine Search

Search Results -

Terms	Documents
(345/519 |709/201 |370/402 |370/463 |370/910 |710/305 |710/2 |710/63 |710/72 |710/100 |710/313 |711/147 |712/32 |712/33).ccls.	7578

Database: US Pre-Grant Publication Full-Text Database; US Patents Full-Text Database; US OCR Full-Text Database; EPO Abstracts Database; JPO Abstracts Database; Derwent World Patents Index; IBM Technical Disclosure Bulletins

Search: L5

Refine Search | Recall Text | Clear | Interrupt

Search History

DATE: Wednesday, November 17, 2004 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
	DB=PGPB,USPT,USOC; PLUR=YES; OP=OR		
L5	710/305,2,63,72,100,313;370/402,463,910;345/519;712/32,33;711/147;709/201.ccls.	7578	L5
	DB=EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR		
L4	L3	0	L4
	DB=PGPB,USPT,USOC; PLUR=YES; OP=OR		
L3	L2 and (port or channel)	37	L3
L2	L1 and ((storage or memory or disk or disc) near5 serial$3)	40	L2
L1	((high adj1 speed) near5 bus) near10 (chipset or chip or IC or (integrated adj1 circuit))	472	L1

END OF SEARCH HISTORY

h e b b cg b e e ch

Refine Search

Search Results -

Terms	Documents
L3 and L5	8

Database:
US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search: L6

Refine Search | Recall Text | Clear | Interrupt

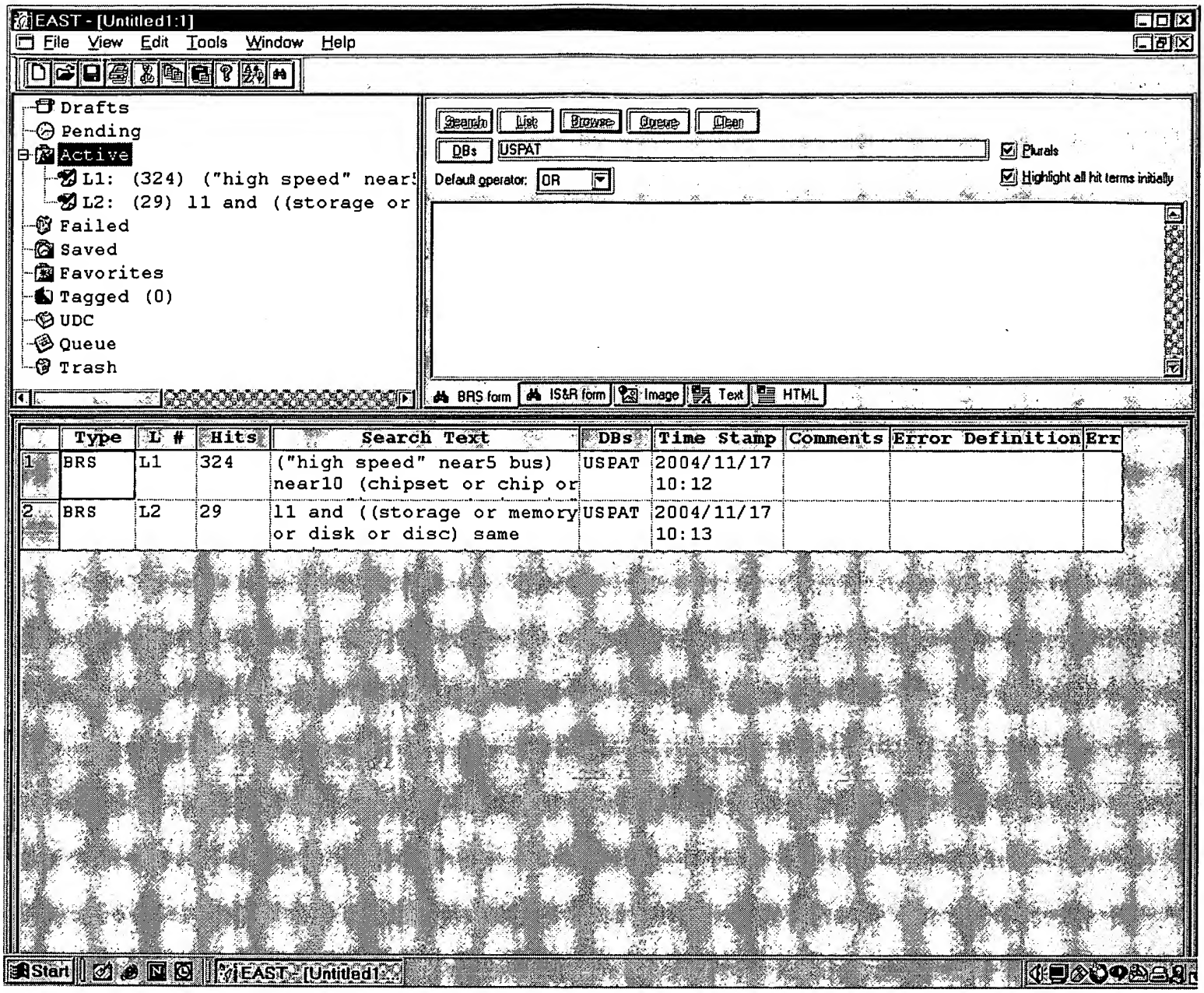
Search History

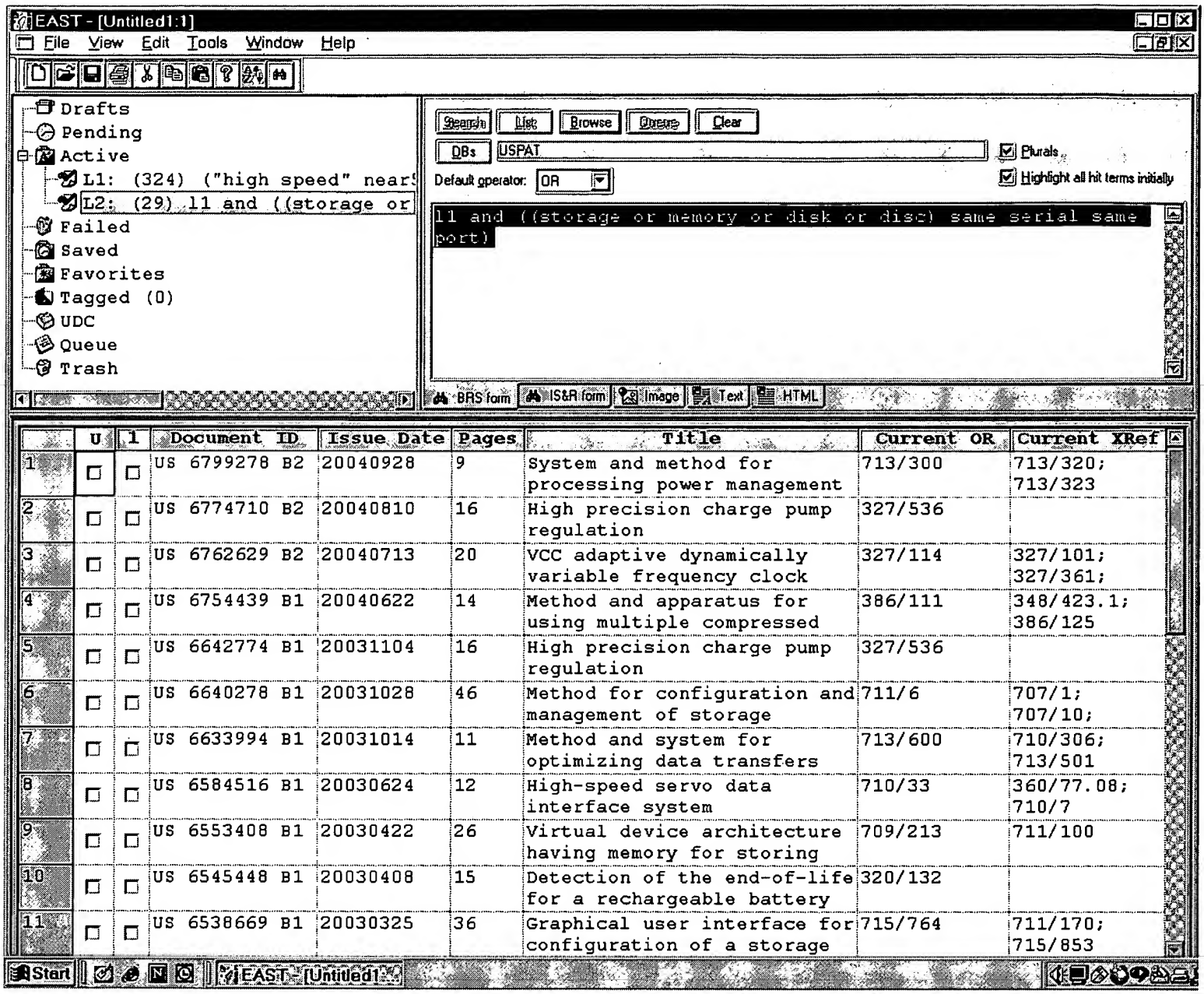
DATE: Wednesday, November 17, 2004 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
	DB=PGPB,USPT,USOC; PLUR=YES; OP=OR		
L6	l3 and L5	8	L6
L5	710/305,2,63,72,100,313;370/402,463,910;345/519;712/32,33;711/147;709/201.ccls.	7578	L5
	DB=EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR		
L4	L3	0	L4
	DB=PGPB,USPT,USOC; PLUR=YES; OP=OR		
L3	L2 and (port or channel)	37	L3
L2	L1 and ((storage or memory or disk or disc) near5 serial$3)	40	L2
L1	((high adj1 speed) near5 bus) near10 (chipset or chip or IC or (integrated adj1 circuit))	472	L1

END OF SEARCH HISTORY

h e b b cg b e e ch


EAST - [Untitled1:1]
File View Edit Tools Window Help
Drafts
Pending
Active
L1: (324) ("high speed" near5
L2: (29) l1 and ((storage or
Failed
Saved
Favorites
Tagged (0)
UDC
Queue
Trash
Search List Browse Queue Clear
DBs USPAT
Plurals
Default operator: OR
Highlight all hit terms initially
BRS form IS&R form Image Text HTML
Type L # Hits Search Text DBs Time Stamp Comments Error Definition Err
1 BRS L1 324 ("high speed" near5 bus) near10 (chipset or chip or USPAT 2004/11/17 10:12
2 BRS L2 29 l1 and ((storage or memory or disk or disc) same USPAT 2004/11/17 10:13
Start
EAST - [Untitled1:1]



	U	1	Document ID	Issue Date	Pages	Title	Current OR	Current XRef
1	☐	☐	US 6799278 B2	20040928	9	System and method for processing power management	713/300	713/320; 713/323
2	☐	☐	US 6774710 B2	20040810	16	High precision charge pump regulation	327/536	
3	☐	☐	US 6762629 B2	20040713	20	VCC adaptive dynamically variable frequency clock	327/114	327/101; 327/361;
4	☐	☐	US 6754439 B1	20040622	14	Method and apparatus for using multiple compressed	386/111	348/423.1; 386/125
5	☐	☐	US 6642774 B1	20031104	16	High precision charge pump regulation	327/536	
6	☐	☐	US 6640278 B1	20031028	46	Method for configuration and management of storage	711/6	707/1; 707/10;
7	☐	☐	US 6633994 B1	20031014	11	Method and system for optimizing data transfers	713/600	710/306; 713/501
8	☐	☐	US 6584516 B1	20030624	12	High-speed servo data interface system	710/33	360/77.08; 710/7
9	☐	☐	US 6553408 B1	20030422	26	Virtual device architecture having memory for storing	709/213	711/100
10	☐	☐	US 6545448 B1	20030408	15	Detection of the end-of-life for a rechargeable battery	320/132	
11	☐	☐	US 6538669 B1	20030325	36	Graphical user interface for configuration of a storage	715/764	711/170; 715/853

Your search matched **6** of **1094442** documents.
A maximum of **500** results are displayed, **15** to a page, sorted by **Relevance Descending** order.

Refine This Search:
You may refine your search by editing the current search expression or enteri new one in the text box.

(high speed) and bus and serial* and port Search

☐ Check to search within this result set

Results Key:
JNL = Journal or Magazine **CNF** = Conference **STD** = Standard

---

1 **A unique application specific MCU for handheld data bank and termi**
Kwok, L.H.; Ho, S.;
Consumer Electronics, IEEE Transactions on , Volume: 35 , Issue: 3 , Aug 19E
Pages:654 - 659

[Abstract] [PDF Full-Text (324 KB)] **IEEE JNL**

---

2 **A parallel, high speed circular queue structure**
Barbour, A.E.; Alhayek, I.;
Circuits and Systems, 1989., Proceedings of the 32nd Midwest Symposium on 16 Aug. 1989
Pages:1089 - 1092 vol.2

[Abstract] [PDF Full-Text (320 KB)] **IEEE CNF**

---

3 **Single event test methodology and test results of commercial gigabi second Fiber Channel hardware**
Carts, M.A.; Marshall, P.W.; Marshall, C.J.; Label, K.A.; Flanegan, M.; Brettha J.;
Nuclear Science, IEEE Transactions on , Volume: 44 , Issue: 6 , Dec. 1997
Pages:1878 - 1884

[Abstract] [PDF Full-Text (736 KB)] **IEEE JNL**

---

4 **A 128Mb multi port media DRAM with four independent 4Gb/s serial ports**
Seunghoon Lee; Kee-Won Kwon; Changhyun Kim; Dongyun Lee; Shin, J.; Soo Cho;
VLSI Circuits, 2004. Digest of Technical Papers. 2004 Symposium on , 17-19 2004

Pages:34 - 35

[Abstract] [PDF Full-Text (272 KB)] **IEEE CNF**

---

5 **EMI issues of universal serial bus and solutions**
Sridhar, K.; Prasad, S.; Punitha, L.; Karunakaran, S.;
Electromagnetic Interference and Compatibility, 2003. INCEMIC 2003. 8th International Conference on , 18-19 Dec. 2003
Pages:97 - 100

[Abstract] [PDF Full-Text (314 KB)] **IEEE CNF**

---

6 **Evaluation of SMT decoupling design in a functioning high-speed PC**
Jun Fan; Knighten, J.L.; Smith, N.W.; Neely, J.;
Electromagnetic Compatibility, 2001. EMC. 2001 IEEE International Symposiu on , Volume: 2 , 13-17 Aug. 2001
Pages:1097 - 1101 vol.2

[Abstract] [PDF Full-Text (424 KB)] **IEEE CNF**

---

h eee e eee g e ch e ch e e c e e e c e

Evaluation of SMT decoupling design in a functionin speed PCB

Jun Fan Knighten, J.L. Smith, N.W. Neely, J.
NCR Corp., San Diego, CA, USA;

This paper appears in: **Electromagnetic Compatibility, 2001. EMC. 2001 International Symposium on**

Meeting Date: 08/13/2001 - 08/17/2001
Publication Date: 13-17 Aug. 2001
Location: Montreal, Que. Canada
On page(s): 1097 - 1101 vol.2
Volume: 2
Reference Cited: 3
Number of Pages: 2 vol. xxx+1381
Inspec Accession Number: 7128977

Abstract:
SMT decoupling capacitor location in DC power **bus** design is a critical design Experimental evaluation of SMT decoupling design is presented in this work fc functioning **high-speed** PCB transmitting 1.0625 Gb/s **serial** data. SMT deco capacitors were removed in several steps while the swept-frequency $|S_{21}|$ an **bus** noise were monitored. It was found that the SMT decoupling capacitors l proximity to the test **ports** decreased $|S_{21}|$ and power **bus** noise at high freq even far above their series resonant frequency. The hardware measurements demonstrate that local decoupling can be beneficial for high-frequency noise r

Index Terms:
capacitors electric noise measurement electromagnetic interference frequency mea monitoring printed circuit design printed circuit testing surface mount technology 1 DC power **bus** design SMT decoupling capacitor location SMT decoupling design h measurements high frequency noise mitigation **high-speed** PCB power **bus** noise r **serial** data transmission series resonant frequency swept-frequency monitoring

Documents that cite this document
There are no citing documents available in IEEE Xplore at this time.

h eee e eee g e ch ch b c be be

Search Results [PDF FULL-TEXT 424 KB] PREV DOWNLOAD CITATION

Home | Log-out | Journals | Conference Proceedings | Standards | Search by Author | Basic Search | Advanced Search | Join IEEE | Web Account | New this week | OPAC Linking Information | Your Feedback | Technical Support | Email Alerting | No Robots Please | Release Notes | IEEE Online Publications | Help | FAQ| Terms | Back to Top

h eee e eee g e ch ch b c be be

Hit List

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Search Results - Record(s) 1 through 8 of 8 returned.

☐ 1. Document ID: US 20030110339 A1

Using default format because multiple data bases are involved.

L6: Entry 1 of 8 File: PGPB Jun 12, 2003

PGPUB-DOCUMENT-NUMBER: 20030110339
PGPUB-FILING-TYPE: new
DOCUMENT-IDENTIFIER: US 20030110339 A1

TITLE: Chip to chip interface for interconnecting chips

PUBLICATION-DATE: June 12, 2003

INVENTOR-INFORMATION:

NAME	CITY	STATE	COUNTRY	RULE-47
Calvignac, Jean Louis	Cary	NC	US	
Heddes, Marco	Cary	NC	US	
Imming, Kerry Christopher	Rochester	MN	US	
Logan, Joseph Franklin	Raleigh	NC	US	
Ozguner, Tolga	Rochester	MN	US	

US-CL-CURRENT: 710/305

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw De

☐ 2. Document ID: US 20030005191 A1

L6: Entry 2 of 8 File: PGPB Jan 2, 2003

PGPUB-DOCUMENT-NUMBER: 20030005191
PGPUB-FILING-TYPE: new
DOCUMENT-IDENTIFIER: US 20030005191 A1

TITLE: Integrated circuit method and apparatus

PUBLICATION-DATE: January 2, 2003

INVENTOR-INFORMATION:

NAME	CITY	STATE	COUNTRY	RULE-47
Montierth, Mark D.	Meridian	ID	US	
Taylor, Richard D.	Eagle	ID	US	

h e b b g e e e f e f ef b e

US-CL-CURRENT: 710/104; 710/313

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw De

---

☐ 3. Document ID: US 6643713 B2

L6: Entry 3 of 8 File: USPT Nov 4, 2003

US-PAT-NO: 6643713
DOCUMENT-IDENTIFIER: US 6643713 B2

TITLE: Apparatus has a microprocessor including DSP and a CPU integrated with each other as a single bus master

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw De

---

☐ 4. Document ID: US 6353863 B1

L6: Entry 4 of 8 File: USPT Mar 5, 2002

US-PAT-NO: 6353863
DOCUMENT-IDENTIFIER: US 6353863 B1

TITLE: Terminal

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw De

---

☐ 5. Document ID: US 6317819 B1

L6: Entry 5 of 8 File: USPT Nov 13, 2001

US-PAT-NO: 6317819
DOCUMENT-IDENTIFIER: US 6317819 B1

TITLE: Digital signal processor containing scalar processor and a plurality of vector processors operating from a single instruction

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw De

---

☐ 6. Document ID: US 5970069 A

L6: Entry 6 of 8 File: USPT Oct 19, 1999

US-PAT-NO: 5970069
DOCUMENT-IDENTIFIER: US 5970069 A

TITLE: Single chip remote access processor

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw De

h e b b g e e e f e f ef b e

☐ 7. Document ID: US 5646651 A

L6: Entry 7 of 8 File: USPT Jul 8, 1997

US-PAT-NO: 5646651
DOCUMENT-IDENTIFIER: US 5646651 A

TITLE: Block mode, multiple access multi-media/graphics memory

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWMC | Draw De

☐ 8. Document ID: US 4907225 A

L6: Entry 8 of 8 File: USPT Mar 6, 1990

US-PAT-NO: 4907225
DOCUMENT-IDENTIFIER: US 4907225 A

TITLE: Data protocol controller

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWMC | Draw De

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Terms	Documents
L3 and L5	8

Display Format: - Change Format

Previous Page Next Page Go to Doc#

h e b b g e e e f e f ef b e

L6: Entry 6 of 8 File: USPT Oct 19, 1999

DOCUMENT-IDENTIFIER: US 5970069 A
TITLE: Single chip remote access processor

Brief Summary Text (7):
In addition, interconnect devices today offer limited or no programmability for end users to customize their applications and minimize network usage costs through optimization or channel utilization.

Brief Summary Text (11):
In one embodiment, the data routing control circuit includes an internal transfer bus, a multi-channel direct memory access (DMA) controller, a central processing unit (CPU), an internal memory, a local memory interface, a local memory controller and a bridge circuit. The internal transfer bus is coupled to the DMA controller, CPU, local memory controller and bridge circuit for passing data, address and control information to and from the various elements. The DMA controller has a first channel coupled to the LAN interface unit, a second channel coupled to the multi-protocol SWAN interface unit and a third channel coupled to the telephony coder-decoder interface unit. The internal memory is coupled to the DMA controller and the CPU for maintaining buffer memory descriptor lists for each DMA channel.

Drawing Description Text (5):
FIGS. 4a-4c are block diagrams illustrating various WAN port configurations of the remote access processor shown in FIG. 3.

Drawing Description Text (6):
FIG. 5 is a block diagram of the WAN port configuration shown in FIG. 4c.

Drawing Description Text (37):
FIG. 61 is a block diagram of a time slot multiplexer and SWAN port configuration within the remote access processor of the present invention.

Drawing Description Text (38):
FIG. 62 is a diagram of a SWAN port configuration register.

Drawing Description Text (39):
FIGS. 63a-63c are port select tables.

Drawing Description Text (47):
FIGS. 89a-90a are diagrams illustrating PCI port control registers.

Detailed Description Text (7):
In alternative embodiments, remote access processor 34 can be located within PC 52 or portable PC 53 to provide LAN and WAN port expansion, which allows connectivity to a diverse set of network interfaces. In FIG. 2b, remote access processor 34 is implemented within an edge router 63. LAN interface 36 is coupled to a plurality of PCs 52a-52c over Ethernet network 50, which allows connectivity to WAN 54 over a variety of SWAN interfaces 40, 42 and 44. Edge router 63 can be located in a stand-alone box, in network server 30 or in any one of the PCs 52a-52c.

Detailed Description Text (8):

FIG. 3 is a block diagram of remote access processor 34. Remote access processor 34 is implemented on a single integrated circuit chip having a plurality of inputs and outputs. In one embodiment, the integrated circuit chip is manufactured with a 3.3 volt, 0.35 micron CMOS fabrication technology and is packaged within a 256 position plastic ball grid array having nominal dimensions of 27 mm.times.27 mm.times.2.1 mm. The communication interfaces include LAN interface 36, PCI interface 38 and SWAN interfaces 40, 42a, 42b and 44. SWAN interface 40 is a multi-protocol SWAN interface. SWAN interfaces 42a and 42b are time division multiplexer (TDM) serial interfaces for supporting two ISDN-BRI networks. SWAN interface 44 is a V.34 coder-decoder (CODEC) interface for coupling to a V.34 CODEC modem. The remaining inputs and outputs of remote access processor 34 include serial peripheral interface (SPI) 55, reference clock input 56, real time clock (RTC) input 57, battery back-up input 58, UART compatible serial I/O port 60, 40 Mhz clock input 62, boundary scan test I/O 64, 32-bit local memory interface 66 and general purpose interrupt/control I/O 70.

Detailed Description Text (9):
V.34 CODEC interface 44 is coupled to V.34 interface controller 72, which provides a serial digital interface to an external V.34 CODEC and digital-to-analog converter. V.34 CODEC interface 44 can also be routed to an external ISDN-BRI channel.

Detailed Description Text (11):
Multi-channel DMA controller 82 is coupled between interface controllers 72, 74 and 76a-76d and memory 84. In a preferred embodiment, memory 84 includes a static random access memory (SRAM). DMA controller 82 is also coupled to internal transfer bus 86, which includes a data bus, an address bus and a control bus. DMA controller 82 has six channels which are coupled to respective controllers 72, 74 and 76a-76d for directing data from any one of the controllers to memory 84 or to transfer bus 86. DMA controller 82 also transfers data from transfer bus 86 and memory 84 to any one of the controllers 72, 74 and 76a-76d. Although any multi-channel DMA controller can be used with the present invention, DMA controller 82 preferably includes a contents addressable memory (CAM) which determines the destination of received data packets based on the address header of the data packet. An example of a DMA controller having a CAM is described in more detail in U.S. Ser. No. 08/761,986, filed Dec. 11, 1996, and entitled DMA CONTROLLER WITH ENHANCED DATA FRAME PROCESSING, which is hereby incorporated by reference.

Detailed Description Text (16):
Memory controller 110 is coupled between local memory interface 66 and internal transfer bus 86. Memory controller 110 is also coupled to DMA controller and bridge circuit 112. Memory controller 110 is an integrated memory controller for supporting various local peripheral memory devices, such as a 32-bit SDRAM or an 8-bit PROM, which may be coupled to local memory interface 66. DMA controller and bridge circuit 112 is coupled between memory controller 110 and PCI interface circuit 114, and has data, address and control buses coupled to internal transfer bus 86. Circuit 112 transfers data packets between PCI interface 114 and local memory, through memory controller 110 and under the control of CPU 90. PCI interface circuit 114 is coupled between PCI interface 38 and internal transfer bus 86. PCI interface 114 is a 33 Mhz, 32-bit (3.3 Volt/5.0 Volt) interface which allows connection to external devices such as a host processor, additional LAN and WAN ports, multiple remote access processors, or network servers, for example. Finally, general purpose interrupt and control circuit 116 is coupled between interrupt and control interface 70 and internal transfer bus 86. In one embodiment, circuit 116 includes twelve programmable, bidirectional pins for serving additional interrupt inputs or control outputs.

Detailed Description Text (20):
DMA controller 82 and CPU 90 maintain at least one "transmit" linked list of buffer memory descriptors (BMDs) in memory 84 for each channel. The BMDs in each list

point to a data packet stored in local memory that are to be transmitted through that DMA channel. DMA controller 82 has a buffer memory descriptor (BMD) pointer register for each channel, which point to the first BMD in the corresponding list. When CPU 90 is notified by memory controller 110 that there is a packet in local memory to be transferred, CPU 90 appends a new BMD to the transmit linked lists of the channel or channels through which the data packet is to be transmitted. CPU 90 then notifies DMA controller 82 that the data packet is ready to be transmitted. Each channel of DMA controller 82 then transfers the packets identified in its transmit linked list to the corresponding controller 72, 74, 76a, 76b, 76c or 76d.

Detailed Description Text (21):
In addition, when the first packet is sent through a particular communication interface, CPU 90 performs a call sequence to setup the desired connection through the corresponding controller. For example, CPU 90 provides SWAN controllers 76a-76d with a telephone number and other data to initiate a call over interface ports 40, 42a and 42b.

Detailed Description Text (25):
When a data packet is received at one of the serial WAN interfaces 40, 42a or 42b or at LAN interface 36, the data packet is passed to DMA controller 82 which temporarily stores the data packets in local memory. CPU 90 maintains at least one, and preferably two, "receive" linked lists in memory 84 for each channel of DMA controller 82. The receive linked list includes a BMD for each received data packet that is stored in local memory and corresponds to that DMA channel.

Detailed Description Text (26):
As each data packet is received by DMA controller 82, the header address is extracted from the data packet and applied to the CAM within DMA controller 82. Based on the header address, the CAM provides an output to DMA controller 82 that identifies the receive linked list or lists to which the data packet should be appended. DMA controller 82 uses the receive linked list to access the BMD which points to the local memory destination of that packet, which is reserved for storing received data packets. DMA controller 82 then moves the data packet into the local memory destination through memory controller 110. In one embodiment, each channel of DMA controller 82 has a corresponding segment of local memory in which to store its received data packets.

Detailed Description Text (27):
When the data packet is in local memory, DMA controller 82 interrupts CPU 90. CPU 90 looks at the receive linked list, determines the packet destination, and if the packet destination is one of the LAN or WAN interfaces, the CPU appends a BMD to the transmit linked list of the channel of DMA controller 82 that corresponds to the destination. The BMD has a buffer address that points to the data packet in local memory. DMA controller 82 then transfers the packet through the corresponding channel and LAN or WAN controller. If the packet destination is a host processor or other device coupled to PCI interface 38, CPU 90 transfers the packet from the local memory through DMA/bridge circuit 112 to PCI interface circuit 114. The DMA portion of circuit 112 has linked list for controlling the transfer of data packets to PCI interface circuit 114. This link list is updated by CPU 90.

Detailed Description Text (28):
When serial data is received at V.34 Codec interface 44, the data represents a sampling of the analog signal transmitted over a public switched telephone network. The sampling is performed by an external V.34 Codec (not shown) coupled to V.34 Codec interface 44. As in the case where serial data is received at one of the LAN or WAN ports DMA controller 82 collects the serial data and stores the data in local memory, through memory controller 110. Once the data samples have been stored in local memory, CPU 90 retrieves the data samples and demodulates the samples into digital data packets by performing a digital filter and data pump function through the V.34 digital signal processing algorithm. CPU 90 then stores the data packets

back into local memory through memory controller 110, and notifies the routing software executed by CPU 90 to look for the packets in local memory. The routing software then retrieves the packets from local memory and determines their destination, as discussed above.

Detailed Description Text (30):
FIGS. 4a-4c are block diagrams illustrating various WAN port configurations of remote access processor 34 shown in FIG. 3. In FIG. 4a, remote access processor 34 is configured to communicate between one Ethernet port and four SWAN ports. In FIG. 4b remote access processor 34 is configured to communicate between one Ethernet port, two ISDN-BRI ports and one V.34 port. In FIG. 4c remote access processor 34 is configured to communicate between one Ethernet port, one multi-protocol SWAN port, one ISDN-BRI port and one V.34 port.

Detailed Description Text (31):
FIG. 5 is a block diagram of an application of remote access processor 34 according to the configuration shown in FIG. 4c. PCI interface 38 is coupled to PCI bus 150, which is coupled to host processor 152. LAN interface 36 is coupled to an Ethernet physical layer device 154, such as a DP83840VCE PHY device which is available from National Semiconductor Corporation. Isolation transformer circuit 156 is coupled between Ethernet physical layer device 154 and LAN network 158 for communication over LAN network 158. Local memory interface 66 is coupled to local memory bus 160, which is coupled to DRAM 162, flash ROM 164, DS-1/E1 Frame Relay framer 166 and Siemens 2186 ISDN transceiver 168. DRAM 162 and flash ROM 164 form the local memory for storing the software algorithms executed by remote access processor 34 and for storing data packets received from the various communications ports of remote access processor 34.

Detailed Description Text (32):
TDM serial interface port 42a is coupled to ISDN transceiver 168. ISDN front end circuit 170 is coupled between ISDN transceiver 168 and ISDN network 172. Similarly, multi-protocol SWAN interface 40 is coupled to framer 166. A DS-1 CSU/E1 isolation transformer circuit 176 is coupled between framer 166 and SWAN network 178. Flash ROM 164 provides 8-bits of instruction data over local memory bus 160. Remote access processor 34 controls the operation of ISDN transceiver 168 and framer 166 using 8-bit bus 174. In an alternative embodiment, in which a Motorola ISDN transceiver is used, SPI interface 55 provides control for the transceiver.

Detailed Description Text (36):
The following example outlines the functionality of the Remote Access Processor (RAP) of the present invention according to one embodiment of the present invention. The RAP provides wide area connectivity to an ethernet LAN, as illustrated in FIG. 2. Up to four WAN ports are supported. WAN technologies and protocols include analog V.34, Basic Rate ISDN, fractional or full DS-1/E1, ADSL, cable modem or other high speed technologies up to the STS-1data rate of 52 Mbps. A PCI interface is provided for server based applications and LAN/WAN port expansion.

Detailed Description Text (65):
PCI.sub.-- INTn (PCI Interrupt): RAP asserts PCI.sub.-- INTn when the Transmit Messaging Mailbox is not empty and the interrupt is enabled in the Primary Port Control Register. Additional information is provided in Section 9.2.3, below, which is entitled "Primary Port Control Registers".

Detailed Description Text (84):
The Serial UART Interface provides a serial port for connecting to a console and allows for RAP system monitor and debug.

Detailed Description Text (142):
BS.sub.-- TRSTn (Test Reset): BS.sub.-- TRSTn, when asserted, resets the Boundary

Scan Test Access Port controller.

Detailed Description Text (157):
The RAP's Central Processing Unit (CPU 90) is based upon LSI Logic Corporation's CW4011 MiniRISC.TM. high performance processor core. CW4011 is the G10 implementation of the LSI Logic CW4010 superscalar MIPS processor. A block diagram of CPU 90 is shown in FIG. 9. CPU 90 interfaces to RAP logic via two interfaces. The SC-bus 220 is a 32-bit bus for accessing PCI, Local Memory and RAP internal registers. The On Chip Access (OCA) 222 bus provides high speed access to the SRAM 84 used as shared memory with the DMA Controller 82. In addition to the CW4011 core, CPU 90 provides: direct-mapped or two way set associative instruction cache, direct-mapped or two-way set associative data cache, a writeback buffer for writeback cache mode, reset control and address translation hardware. The RAP Chip and CW4011 core are configured such that all data structures are big endian.

Detailed Description Text (159):
In addition to the CW4011 registers, CPU 90 contains the registers identified in FIG. 10. CPU 90 directly accesses external devices on either the PCI or Local Memory Bus. The slave address space reflected on the PCI is 128 Mbytes and, together with Internal Memory SRAM 84 and hardware registers results in internal address buses [31:0]. Address bits [31:30] define whether the operation is a PCI bus, local memory, internal SRAM 84 or hardware registers. The CPU.sub.-- AddrMap register holds the most significant bits of an address for an CPU access through the PCI or Local Memory Port in order to extend external address buses to 32 bits.

Detailed Description Text (162):
The 5-bit wide field, CPU.sub.-- PciMSB, is concatenated with the 27 bit wide internal address to form 32 bit wide external address on the PCI bus. The 5-bit wide field, CPU.sub.-- LmMSB, is concatenated with the 27 bit wide internal address to form 32 bit wide external address on the Secondary Port.

Detailed Description Text (164):
FIG. 12 illustrates the PCI and Local Memory address formation. The exception vector space located at virtual address 0xbfc0 0000 is further remapped using the field CPU.sub.-- ExcMap as shown in FIG. 13. This space may be mapped on the Secondary Port for booting from byte wide EPROM or SRAM/DRAM code downloaded from PCI. FIG. 13 illustrates the Local Memory address formation for exception vectors.

Detailed Description Text (174):
Looking at FIG. 15, Int.sub.-- DMA is the buffer completion/exception interrupt from the DMA controller; Int.sub.-- EN is an exception interrupt from the ethernet controller; Int.sub.-- V.34 triggers the V.34 datapump; the Int.sub.-- PCI interrupt is asserted whenever a PCI error has occurred, to report PCI related Mailbox interrupts or to signal PCI DMA completion; Int.sub.-- SWANn indicate an exception by the associated SWAN controller; Int.sub.-- RTC is the Real Time Clock interrupt; an enabled Timer time-out event generates the Int.sub.-- Timer interrupt; Int.sub.-- GPI/O are external interrupts from the General Purpose I/O port; and Int.sub.-- UART is the UART interrupt.

Detailed Description Text (176):
The Local Memory Controller (LMC) 110 provides access to external memory used for packet buffers, CPU instructions and data structures Synchronous DRAM devices and SGRAM used as an SDRAM are supported. The LMC is also used to access byte-wide EPROM for CPU boot and the control port of external PHY devices.

Detailed Description Text (179):
There are 4 bus masters that access the local memory: the CPU, the multi channel DMA engine, an external PCI bus master, and the PCI DMA engine. Priority will be a simple round robin between the CPU, the multi channel DMA and one of the PCI masters. The PCI bus master and the PCI DMA engine will have a round robin priority

between them to see who gets to arbitrate for memory. The CPU is the only master that can access the PHY page. The CPU and the PCI masters can access the EPROM, but the multi channel DMA cannot. Burst transfers are not allowed to the EPROM or PHY pages.

Detailed Description Text (207):
The EPROM interface is provided for boot code. The CPU 90 and the PCI external master can access the EPROM page. The port is only 8 bits so the LMC 110 will convert word accesses to 4 byte accesses transparent to the CPU 90 or PCI master. Byte accesses are also allowed. The RAP is big endian, so address 3 will go to the least significant byte of a word transfer, address 0 will go to the most significant byte. A byte read of address 3 will have data valid on the least significant byte of data. A byte read of address 0 will have data valid on the most significant byte of data. Even though the port is 8 bits, the LMC will move data to the proper byte lane as seen by the CPU and PCI.

Detailed Description Text (219):
The PHY interface is provided to allow access to 8 bit chips. The CPU is the only master that can access the PHY page. The port only allows byte transactions. Any word transaction will complete on the internal chip bus but reads will always return 0 and nothing will be written on a write. The address and data may be muxed or not muxed. The control signals for this port are CSn, OEn, WEn, and ALE. The timing is governed by a programmable wait state generator. No ready or acknowledge signals will be checked. If there are multiple devices on the PHY port then an external PAL will be needed to generate the chip selects. An external PAL will probably be needed to convert the timing of the CSn, OEn, WEn, and ALE signals to what the device needs.

Detailed Description Text (235):
The DMAC 82 supports 6 transmit and 12 receive link list buffers. Transmit buffers are assigned on a per channel basis, whereas receive buffers are assigned based on channel # and frame address. Up to 6 bytes of frame address can be used to determine receive buffer, or if the frame should be discarded.

Detailed Description Text (236):
DMAC arbitration for the RAP internal bus 86 alternates between input and output. Arbitration for channel service uses a two priority method based on communication channel FIFO status. Each serial port has programmable thresholds. Those channels with data levels exceeding threshold on receive, or below threshold during transfer are given higher priority than other channels. This allows the DMAC 82 to adapt to peaks in data transfer rate and allows high speed and low speed channels to peacefully coexist.

Detailed Description Text (238):
For multiplexed operations, Buffer Memory Control sections 254 and 256 govern which multiplexed channel is transmitting and receiving from or to the Local Memory Interface over bus 86. In order to prevent overruns and underruns on any particular channel, each channel is given an equal time share to access Buffer Memory. This process is achieved in DMAC 82 by using word groups.

Detailed Description Text (240):
In the transmit direction, one Transmit Data Queue 250 is required per multiplexed channel. The Channel Number (Chan #) is an input to the Transmit Word Group Queue and selects the appropriate queue in which to write transmit data into.

Detailed Description Text (241):
Buffer Memory Control 254 will transfer one word group per DMA channel from Buffer Memory into the Transmit Word Group Queue 250. The specified Channel Number (Chan #) accompanies each data word through the Transmit Word Group Queue 250. Four Byte Enable flags also accompany each data word through the Transmit Word Group Queue

250.

Detailed Description Text (244):
In the receive direction, each channel of DMAC 82 has an associated byte wide Receive Data Queue within circuit 252. In additicn to the 8 bits of data, the Receive Data Queues have 1 bit indicating the End of Frame (EOF). The EOF bit is set in the Receive Data Queue byte location immediately following the last data byte of a frame. Bits 0-7 of the byte associated with the EOF bit are used as error flags from the serial interface controller.

Detailed Description Text (246):
A Byte To Word Group Collection circuit within circuit 252 will get a byte of data from the Receive Data Queue. The Byte To Word Group Collection circuit will collect data bytes for all enabled channels. When a word group is reached, the Byte To Word Group Collection circuit will notify Buffer Memory Control 256, and the word group will be written to Buffer Memory through output 258.

Detailed Description Text (247):
While a word group is being transferred to Buffer Memory, Byte To Word Group Collection will continue to collect data bytes for all enabled channels.

Detailed Description Text (248):
Circuit 252 further includes a Header Offset Lookup Table. The Header Offset Lookup Table provides a parameter for indexing the beginning of the receive frame header on a per channel (RxQue channel) basis. This 4-bit parameter is an integer offset into the incoming frame to indicate the byte number of the first header byte. If the first byte of the frame is the beginning of the Frame Header field, then Header Offset parameter is set to zero.

Detailed Description Text (249):
Circuit 252 includes a Receive Frame Header Array, which is a small memory array used for incoming (receive) data frames. Four bytes of information are stored per incoming channel number. For each frame received, the DMAC captures the 4 byte Frame Header field and then routes the field to the Receive Frame Action CAM 260. The Header Offset parameter defines the absolute offset to the Frame Header field in the receive data frame.

Detailed Description Text (252):
As mentioned above, DMAC 82 and CPU 90 maintain one Linked Buffer Memory Descriptor (BMD) List in SRAM 84 for each transmit channel and two linked BMD lists for each receive channel of DMAC 82. Configuration Registers/Processor Interface 270 includes one BMD pointer register for each BMD list maintained in SRAM 84, which points to the addresses of the active transmit or receive buffer memory descriptor in SRAM 84. These descriptors provide the transmit and receive parameters for the data presently being transmitted and received. The BMD pointer registers are updated by DMA controller 82.

Detailed Description Text (261):
RxErr[7:0] are the receive error flags from the Serial Ports. The Error flag definition from the ethernet port 36 is shown in FIG. 26. RSV22 indicates a carrier event was detected some time after the last receive frame. A carrier event is defined as activity on the channel that does not result in a packet receive attempt. An example is receiving a preamble, but no SFD, or receiving more than 7 bytes of preamble. RSV20 indicates a bad packet is received. RSV19 indicates a long event was previously seen. RSV18 indicates an invalid preamble (not 55hex or code 5055hex was received with this frame). RSV15 indicates a CRC error was detected with the receive frame. RSV14 indicates a dribble nibble (odd number of nibbles) was received in is this frame. RSV13 indicates a 4B/5B code violation was detected on receive.

Detailed Description Text (262):
The Error flag definition from the SWAN ports is shown in FIG. 27. CRC.sub.-- Err indicates a CRC error was detected for synchronous protocols or a parity error was detected for asynchronous protocol. Frame.sub.-- Err indicates the HDLC trailing flag was received off byte boundary or a missing stop bit was detected in asynchronous mode. Abort.sub.-- Err indicates the HDLC frame ended with an abort or Data Carrier Detect was deasserted prior to the trailing HDLC flag in synchronous mode. For async, Abort.sub.-- Err indicates detection of a break condition or the Deassertion of DCD.

Detailed Description Text (264):
DMAC 82 maintains several registers in block 270. These registers include a DMAC Control Register, a DMAC Transmit Channel Enable Register, a DMAC Receive Channel Enable Register, a DMAC Interrupt Enable Register, a DMAC Interrupt Queue Register, a DMAC Header Offset Lookup Table, Receive Frame Action CAM Compare Data Registers 1 and 2, a Receive Frame Action CAM Compare Mask Register, and a Receive Frame Action Register. These registers are illustrated in FIGS. 28a-38a, and individual bits are defined in FIGS. 28b-38b, respectively.

Detailed Description Text (267):
The LSI Logic V.34 SoftModem core is a software function ported to the RAP chip. All of the V.34 modulaticn/demodulation and call control intelligence is contained in the V.34 software available trom LSI Logic Corporation. This software, however, needs certain ports to communicate with the logic and access circuitry that is Implemented external to the chip in order to drive the telephone line. The RAP chip provides an integrated set of controls and interface ports designed to connect directly to preferred codec/DAA circuitry located external to the RAP. This integrated set of controls and interface ports is referred to as the "V.34 Interface Block 72".

Detailed Description Text (284):
Four SWAN Controllers 76a-76d (FIG. 3) provide multiprotocol framing for up to four WAN ports. Two or three SWAN controllers may be required for an ISDN BRI connection for 2B or 2B+D configurations, respectively, at TDM interfaces 42a and 42b. Maximum data rates are protocol dependent. Clocked synchronous connections have a maximum data rate of 52 Mbps. Non-clocked synchronous connections are supported up to 2.048 Mbps, while asynchronous (start/stop) connections are limited to 500 Kbps.

Detailed Description Text (297):
The SWAN Memory Map 320 is shown in FIG. 43, and includes several registers. SWAN Configuration Register 320a is shown in FIGS. 44a and 44b, SWAN Baud Rate Generator Register 320b is shown in FIGS. 45a and 45b, SWAN Reset/Halt Register 320c is shown in FIGS. 46a and 46b, SWAN Status Register 320d (bits 15-0) is shown in FIGS. 47a-47c, SWAN Status Register 320d (bits 31-16) is shown in FIGS. 48a and 48b, SWAN Sync Character Register 320e is shown in FIGS. 49a and 49b, SWAN Idle Character Register 320f is shown in FIGS. 50a and 50b, SWAN Interval Timer Register 320g is shown in FIGS. 51a and 51b, SWAN Idle Count Register 320h is shown in FIGS. 52a and 52b, SWAN Channel Configuration Register 320i is shown in FIGS. 53a-53c, SWAN Channel Pacing Register 320j is shown in FIGS. 54a and 54b, SWAN Rate Indication Register 320k is shown in FIGS. 55a and 55b, and SWAN FIFO Threshold Register 3201 is shown in FIGS. 56a and 56b.

Detailed Description Text (304):
4.8 Time Slot Multiplexer & WAN Port Configuration

Detailed Description Text (305):
FIG. 61 is a block diagram which illustrates WAN port configurability. Two Time Slot Multiplexers (TSMs) 340 and 342 provide TDM highways for ISDN BRI, channelized T1/Elland other TDM applications. The four SWAN controllers 76a-76d and two Serial Peripheral Interfaces (SPI) 80 connect to both TSMs. Serial Channel to time slot

mapping is programmed in the Switch Table RAM associated with each TSM. Alternately, SPI #1 may be mapped to external signal pins as a general purpose serial port.

Detailed Description Text (307):
WAN Port configuration is programmable in the WAN Port Conf iguration Register, which is shown in FIG. 62. SPI1SEL[1:0] (SPI 1 Port Select bits 11-10) select which WAN Port SPI #1 is connected to as defined in FIG. 63a. SPI2SEL[1:0] (SPI 2 Port Select bits 9-8) select which WAN Port SPI #2 is connected to as defined in FIG. 63b. SWANnSEL[1:01] (SWAN Port Selects bits 7-0) select which WAN Port each SWAN is connected to as defined in FIG. 63c.

Detailed Description Text (310):
CHEN Channel Enable bit 15

Detailed Description Text (311):
Channel Enable, when asserted, enables the channel designated by CH#[4:0] onto the transmit TDM bus for the time slot duration specified by CNT[7:0]. When CHEN is clear, no channel is enabled on the transmit TDM bus. The value of CHEN is available on core output signal CHN.sub.-- EN for use in tristate control of the Transmit TDM interface pin. For receive, the CHEN pin acts as an enable to the decode of channel selects.

Detailed Description Text (316):
CH#[2:0] Channel Number bits 12-8

Detailed Description Text (317):
CH#[2:0] identifies the physical channel assigned to the time slot.

Detailed Description Text (345):
4.9.3 Port Control Registers

Detailed Description Text (347):
For most applications, RAP 34 is connected to PCI systems that may have high latency. The PCI DMA Controller 112 (FIG. 3) optimizes performance by allowing long bursts. PCI burst length may need to be fine tuned with Local Memory Bus burst length to achieve optimum system performance. The PCI Interface 114 and Local Memory controller 110 integrate maximum burst timers which are controlled by the PCI.sub.-- Ctrl register. The Primary Port maximum burst size timer works independently from the PCI latency timer. The main purpose of these maximum burst size counters is to break up long packet transfers so the comm DMA channels do not stall.

Detailed Description Text (355):
The mailbox register is shown in FIG. 98. The signal MbxTxEmpty is used to generate the PCI.sub.-- INTn signal. The PCI.sub.-- INTn is asserted ashen enabled in the Primary Port Control Register and the transmit mailbox is not empty. An external controller is expected to isolate the mailbox register from events that cause overflow of the Fifo. If an external bus master on the PCI bus reads an empty Transmit FIFO 350, value 0 is returned. If the PCI master attempts to write to a full Receive FIFO 352, the written data is dropped and a non-vectored interrupt is asserted to the CPU. The external master is expected to prevent overflow of the mailbox register. If the Rx FIFO 352 overflows, the interrupt MxRxOVRFL is sent to the CPU 90. The CPU 90 sends a message to the PCI master indicating that the FIFO overflowed. The Mailbox status signals are visible to the PCI bus master in the PCI.sub.-- Control register. If the CPU 90 reads an empty Receive FIFO 352, value 0 is returned. If the CPU 90 attempts to write into a full Transmit FIFO 350, the CPU is stalled until the FIFO becomes non-empty and the interrupt MbxTxOVRFL is generated. The signals MbxRxFull, MbxRxTHLD and MbxRxEmpty generate a vectored interrupt to the CPU 90.

Detailed Description Text (365):
The Buffer Memory Descriptor (BMD), shown in FIG. 100, is 16 bytes long and resides in Buffer Memory, aligned on an 16 byte boundary. Word 0, bits 0-3 and 8-12, are the BMD Flags. These flags are not used in the PCI DMA engine. They may be used in the n-channel DMA controller. Word 0, bits 4-7 and 13-31 are not used and may be reserved for future use. Word 1, bits 16-31, is the Byte Count field for this data buffer and specifies the number of bytes that are to be transferred to or from this data buffer. Word 1, bits 0-15 are not used. Word 2, bits 0-31 contains the Buffer Memory Address of this data buffer. The data buffer can start on any address byte boundary. Word 3, bits 0-31 contains a Next BMD Pointer. This is the Buffer Memory address of the next BMD in the linked list. Since a BMD must start on a 16 byte address boundary, bits 0-3 of the Next BMD Pointer are expected to be zero. The end of the linked BMD list is reached when the Next BMD Pointer is zero.

Detailed Description Text (374):
The General Purpose I/O port provides 12 signals that are programmable as inputs or outputs. The outputs are directly mapped to the CPU register space and are useful as control signals to external devices, modem control signals, or LED drivers. As inputs, the General Purpose I/O pins may be used as CPU vectored interrupts.

Current US Original Classification (1):
370/402

CLAIMS:

1. A single chip remote access processor for receiving and transmitting data packets having headers with destination addresses, the processor comprising:

a plurality of communication interface units adapted to transmit and receive the data packets and comprising a local area network (LAN) interface unit, a first multi-protocol serial wide area network (SWAN) interface unit, a telephony coder-decoder interface unit and a peripheral component interface (PCI) unit;

a multi-channel direct memory access (DMA) controller which is coupled to the plurality of communication interface units and comprises a plurality of channels and a contents addressable memory (CAM), wherein each channel comprises a transmit queue and a receive queue and wherein the CAM comprises a contents compare input coupled to the receive queues so as to receive the destination address of each received data packet and comprises a compare output that identifies which of a plurality of receive linked lists each data packet received by the receive queues is to be appended based on a comparison of the respective destination address;

a central processing unit (CPU) is adapted to maintain a transmit linked list of transmit buffer memory descriptors (BMDs) for each channel of the DMA controller and to maintain the plurality of receive linked lists of receive BMDS, wherein each transmit BMD corresponds to a data packet to be transmitted through the respective channel and wherein each receive BMD corresponds to a data packet received through one of the channels.

5. The single chip remote access processor of claim 2 wherein:

the plurality of communication interface units further comprises second, third and fourth multi-protocol SWAN interface units; and

the plurality of channels of the DMA controller comprises a first channel coupled to the LAN interface unit, a second channel coupled to the multi-protocol SWAN interface unit, a third channel coupled to the telephony coder-decoder interface unit, a fourth channel coupled to the second multi-protocol SWAN interface unit, a fifth channel coupled to the third multi-protocol SWAN interface unit and a sixth

channel coupled to the fourth multi-protocol SWAN interface unit.

6. The single chip remote access processor of claim 5 and further comprising:

a time division multiplexer interface port; and

a time division multiplexer coupled between the time division multiplexer interface port and the first, second, third and fourth multi-protocol SWAN interface units.

7. The single chip remote access processor of claim 6 and further comprising a multi-protocol SWAN interface port coupled to the first multi-protocol SWAN interface unit.

14. A remote access processor comprising:

an internal transfer bus;

a plurality of communication interface units comprising a local area network (LAN) interface unit, a first multi-protocol serial wide area network (SWAN) interface unit, a telephony coder-decoder interface unit and a peripheral component interface (PCI) unit, wherein the telephony coder-decoder interface unit comprises a serial interface for receiving modulated digital samples;

a multi-channel direct memory access (DMA) controller coupled to the internal transfer bus and having a first channel coupled to the LAN interface unit, a second channel coupled to the multi-protocol SWAN interface unit and a third channel coupled to the telephony coder-decoder interface unit;

a central processing unit (CPU) coupled to the internal transfer bus which is adapted to receive a first data packet through the first or second channel of the DMA controller, modulate the first data packet for transmission through the telephony coder-decoder interface unit and route the modulated first data packet to the telephony coder-decoder interface unit, through the third channel of the DMA controller and is adapted to receive the modulated digital samples from the telephony coder-decoder through the third channel of the DMA controller, demodulate the digital samples, translate the demodulated digital samples into a second data packet, and route the second data packet to the LAN interface unit or the multi-protocol SWAN interface unit, through the first and second channels, respectively, of the DMA controller;

a memory coupled to the DMA controller and the CPU; and

wherein the PCI interface unit is coupled to the internal transfer bus.

15. A method of routing a data packet in a circuit comprising a plurality of communication interfaces and a direct memory access (DMA) controller having multiple channels, comprising:

coupling the plurality of communication interfaces to corresponding channels of the DMA controller;

receiving the data packet at a first of the plurality of communication interfaces;

passing the data packet to the DMA channel that is coupled to the first communication interface;

storing the data packet in a buffer memory;

maintaining a plurality of receive linked buffer memory description (BMD) lists;

maintaining a transmit linked BMD list for each channel of the DMA controller;

determining a destination of the data packet;

appending a buffer memory descriptor (BMD), which corresponds to the data packet, to at least one of the receive linked BMD lists, as a function of the destination;

appending the BMD to the transmit linked BMD list of a second channel of the DMA controller as a function of the destination;

passing the data packet from the buffer memory to the second channel; and

transmitting the data packet from the second channel to the communication interface that is coupled to the second channel.

16. The method of claim 15 wherein the step of maintaining a plurality of receive linked BMD lists comprises maintaining a plurality of receive linked BMD lists for each channel of the DMA controller.

17. The method of claim 16 wherein the step of maintaining a plurality of receive linked BMD lists comprises maintaining two receive linked BMD lists for each channel of the DMA controller.

18. The single chip remote access processor of claim 1 wherein the CPU comprises means for maintaining a plurality of receive linked BMD lists for each channel of the DMA controller.

Previous Doc Next Doc Go to Doc#

L6: Entry 6 of 8 File: USPT Oct 19, 1999

US-PAT-NO: 5970069
DOCUMENT-IDENTIFIER: US 5970069 A

TITLE: Single chip remote access processor

DATE-ISSUED: October 19, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Kumar; Shailendra	San Jose	CA		
Sonnek; Christopher D.	North St. Paul	MN		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
LSI Logic Corporation	Milpitas	CA			02

APPL-NO: 08/ 840575 [PALM]
DATE FILED: April 21, 1997

INT-CL: [06] H04 L 12/28, H04 J 3/16

US-CL-ISSUED: 370/402; 370/401, 370/466
US-CL-CURRENT: 370/402; 370/401, 370/466

FIELD-OF-SEARCH: 370/351, 370/355, 370/357, 370/360, 370/389, 370/401, 370/402, 370/465, 370/466, 370/463, 370/467

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected Search ALL Clear

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	5610910	March 1997	Focsaneanu	370/401
☐	5640399	June 1997	Rostoker	370/402

ART-UNIT: 272

PRIMARY-EXAMINER: Patel; Ajit

ASSISTANT-EXAMINER: Pizarro; Ricardo M.

h e b b g e e e f c e f e ge

ATTY-AGENT-FIRM: Westman, Champlin & Kelly, P.A.

ABSTRACT:

A single chip integrated remote access processor circuit has a plurality of communication interface units, including a local area network (LAN) interface unit, a first multi-protocol serial wide area network (SWAN) interface unit, a telephony coder-decoder interface unit and a peripheral component interface (PCI) unit. A data routing control circuit is coupled to the plurality of communication interface units for controlling data transfer between the interface units.

18 Claims, 185 Drawing figures

Previous Doc Next Doc Go to Doc#

L6: Entry 6 of 8 File: USPT Oct 19, 1999

US-PAT-NO: 5970069
DOCUMENT-IDENTIFIER: US 5970069 A

TITLE: Single chip remote access processor

DATE-ISSUED: October 19, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Kumar; Shailendra	San Jose	CA		
Sonnek; Christopher D.	North St. Paul	MN		

ASSIGNEE-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY	TYPE CODE
LSI Logic Corporation	Milpitas	CA			02

APPL-NO: 08/ 840575 [PALM]
DATE FILED: April 21, 1997

INT-CL: [06] H04 L 12/28, H04 J 3/16

US-CL-ISSUED: 370/402; 370/401, 370/466
US-CL-CURRENT: 370/402; 370/401, 370/466

FIELD-OF-SEARCH: 370/351, 370/355, 370/357, 370/360, 370/389, 370/401, 370/402, 370/465, 370/466, 370/463, 370/467

PRIOR-ART-DISCLOSED:

U.S. PATENT DOCUMENTS

Search Selected Search ALL Clear

	PAT-NO	ISSUE-DATE	PATENTEE-NAME	US-CL
☐	5610910	March 1997	Focsaneanu	370/401
☐	5640399	June 1997	Rostoker	370/402

ART-UNIT: 272

PRIMARY-EXAMINER: Patel; Ajit

ASSISTANT-EXAMINER: Pizarro; Ricardo M.

h e b b g e e e f c e f e ge

ATTY-AGENT-FIRM: Westman, Champlin & Kelly, P.A.

ABSTRACT:

A single chip integrated remote access processor circuit has a plurality of communication interface units, including a local area network (LAN) interface unit, a first multi-protocol serial wide area network (SWAN) interface unit, a telephony coder-decoder interface unit and a peripheral component interface (PCI) unit. A data routing control circuit is coupled to the plurality of communication interface units for controlling data transfer between the interface units.

18 Claims, 185 Drawing figures

Previous Doc Next Doc Go to Doc#

US006427216B1

(12) **United States Patent**
El-Kik et al.

(10) **Patent No.: US 6,427,216 B1**
(45) **Date of Patent: Jul. 30, 2002**

(54) **INTEGRATED CIRCUIT TESTING USING A HIGH SPEED DATA INTERFACE BUS**

(75) Inventors: **Tony S. El-Kik, Lehigh; Jeffrey P. Grundvig, Macungie, both of PA (US)**

(73) Assignee: **Agere Systems Guardian Corp., Orlando, FL (US)**

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/265,933**

(22) Filed: **Mar. 11, 1999**

(51) Int. Cl.[7] G06F 11/00
(52) U.S. Cl. 714/724; 714/724
(58) Field of Search 714/726, 724, 714/30, 31, 27

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,355,369 A 10/1994 Greenberger et al. 371/22.3
5,668,815 A * 9/1997 Gitzinger et al. 714/719
5,682,391 A 10/1997 Narayanan 371/22.3

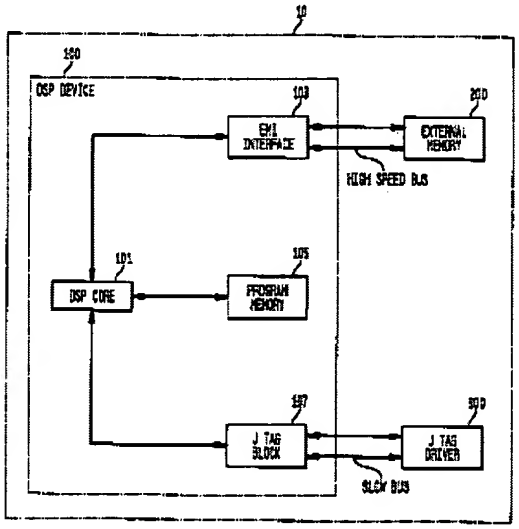
* cited by examiner

Primary Examiner—Phung M. Chung
(74) *Attorney, Agent, or Firm*—Synnestvedt & Lechner LLP

(57) **ABSTRACT**

The use of a JTAG port for boundary scan testing of integrated circuits, (IC) thereby allowing for the testing of the IC's after they have been mounted onto a circuit board. The present invention speeds the testing of integrated circuitry by introducing an external memory where all the test vectors are stored. This external memory is connected to the digital processor core by a high speed interface extended memory interface (EMI). The test vectors are uploaded into the digital processor core from the external memory via the high speed EMI interface.

19 Claims, 5 Drawing Sheets



US-PAT-NO: 6427216

DOCUMENT-IDENTIFIER: US 6427216 B1

TITLE: Integrated circuit testing using a high speed data interface bus

---------- KWIC ---------

TITLE - TI (1):
Integrated circuit testing using a high speed data interface bus

Brief Summary Text - BSTX (19):
According to the principles of the present invention, a method and an apparatus for testing an integrated circuit by the use of a high speed data interface bus is provided. In the present invention, an integrated circuit having a programmable digital processor, a program (internal) memory, and a JTAG driver is coupled to an external memory via a high speed (parallel) interface bus. The external memory stores a plurality of test vectors to be used for testing operation of the integrated circuit, and the high speed interface bus is used to download these test vectors. The JTAG circuit is used to download the control program controlling the download of the test vectors.

Brief Summary Text - BSTX (21):
The proposed method and apparatus reduces the testing time by almost 50% because only a short program (control program) is downloaded via the slow serial port. The actual test programs are downloaded via a high speed interface bus, for example, a high speed external memory interface.

Claims Text - CLTX (8):
8. The integrated circuit of claim 7, wherein said JTAG driver further comprises: a test data register (TDR) having a serial test input port and having n-bit locations for serially receiving a control program through said serial test input port, and for transferring said control program in parallel into said program memory via said digital processor; a test control register (JCON) for initiating downloading and execution of said control program in order to download said test vectors and produce test results; and wherein the test results of said test vectors are transferred via said programmable digital processor into said test data register for serial transfer through said serial test output port.

US006282592B1

(12) **United States Patent**
Aston et al.

(10) Patent No.: **US 6,282,592 B1**
(45) Date of Patent: **Aug. 28, 2001**

(54) **METHOD AND APPARATUS FOR HIGH-SPEED DATA TRANSMISSION BUS ENTRAINMENT**

(75) Inventors: **Robert Allan Aston**; **James Leslie Watchorn**; **James C. McLaughlin**, all of Nepean (CA)

(73) Assignee: **Nortel Networks Limited**, St. Laurent (CA)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/210,836**

(22) Filed: **Dec. 14, 1998**

(51) Int. Cl.[7] G06F 13/00

(52) U.S. Cl. 710/100; 710/126; 714/30

(58) Field of Search 710/100, 126, 710/131; 711/201; 714/30

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,055,997 * 10/1991 Sindler et al. .
5,548,533 * 12/1996 Randell et al. 340/195
5,701,330 * 6/1998 Funahashi et al. 710/15
5,822,534 * 10/1998 Whittaker et al. 714/703

OTHER PUBLICATIONS

U.S. application No. 08/997,777, John Gordon Hogeboom, filed Dec. 12, 1997.
U.S. application No. 09/002,113, John Gordon Hogeboom, filed Dec. 31, 1997.

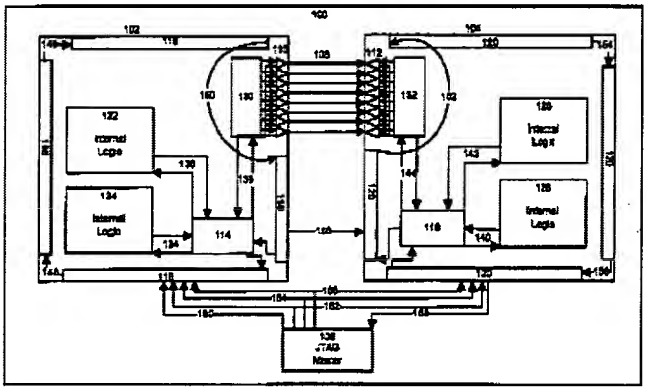
* cited by examiner

Primary Examiner—Ario Etienne

(57) **ABSTRACT**

The present invention relates to a control device, for use on an electronic circuit board, to perform high-speed data transmission bus entrainment. The control device includes a signal processing component acquiring either one of two modes of operation, specifically a diagnostic mode and a bus entrainment diagnostic mode. When in diagnostic mode, the signal processing component performs a diagnostic procedure on the electronic circuit board. When in bus entrainment mode, the signal processing component performs a bus entrainment procedure.

18 Claims, 5 Drawing Sheets



Data Bus Cell

US005970069A

United States Patent [19]

Kumar et al.

[11] Patent Number: 5,970,069

[45] Date of Patent: Oct. 19, 1999

[54] **SINGLE CHIP REMOTE ACCESS PROCESSOR**

[75] Inventors: Shailendra Kumar, San Jose, Calif.; Christopher D. Sunnak, North St. Paul, Minn.

[73] Assignee: LSI Logic Corporation, Milpitas, Calif.

[21] Appl. No.: 08/840,575

[22] Filed: Apr. 21, 1997

[51] Int. Cl.[6] H04L 12/28; H04J 3/16

[52] U.S. Cl. 370/402; 370/401; 370/466

[58] Field of Search 370/351, 355, 370/357, 360, 389, 401, 402, 465, 466, 463, 467

[56] References Cited

U.S. PATENT DOCUMENTS

5,610,910 3/1997 Focsaneanu 370/401

5,640,399 6/1997 Rostoker 370/402

Primary Examiner—Ajit Patel

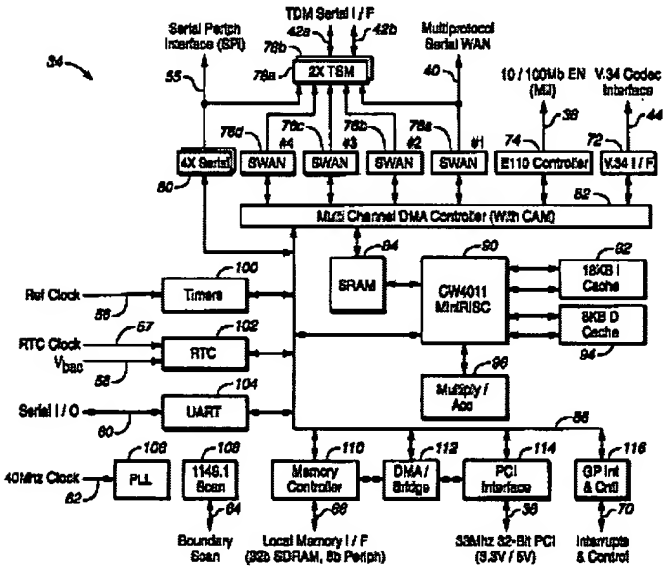
Assistant Examiner—Ricardo M. Pizarro

Attorney, Agent, or Firm—Wagner, Champlin & Kelly, P.A.

[57] ABSTRACT

A single chip integrated remote access processor circuit has a plurality of communication interface units, including a local area network (LAN) interface unit, a first multi-protocol serial wide area network (SWAN) interface unit, a telephony coder-decoder interface unit and a peripheral component interface (PCI) unit. A data routing control circuit is coupled to the plurality of communication interface units for controlling data transfer between the interface units.

18 Claims, 83 Drawing Sheets



US005517626A

United States Patent [19]

Archer et al.

[11] Patent Number: 5,517,626

[45] Date of Patent: May 14, 1996

[54] **OPEN HIGH SPEED BUS FOR MICROCOMPUTER SYSTEM**

[75] Inventors: Jordan J. Archer; Ajit J. Deora, both of San Jose; Kent S. Leung, Milpitas; Leon Peng, Mountain View; Robert C. Schoenmeyer, Los Altos; David J. Scott, Gilroy; Sanjay Sharma, Sunnyvale; Virgil Stevens, Rancocas, all of Calif.

[73] Assignee: 83, Incorporated, Santa Clara, Calif.

[21] Appl. No.: 11,449

[22] Filed: Jan. 29, 1993

Related U.S. Application Data

[62] Division of Ser. No. 511,043, May 7, 1990, abandoned.

[51] Int. Cl.[6] G06F 13/00

[52] U.S. Cl. 395/290; 395/306; 395/445; 395/285

[58] Field of Search 395/325, 725, 395/425, 275, 290, 306, 457, 445, 853, 733, 735

[56] References Cited

U.S. PATENT DOCUMENTS

4,152,782	2/1979	Szmidt et al.	395/275
4,543,628	9/1985	Pomfret	395/275
4,594,653	6/1986	Iwashita et al.	395/800
4,594,660	6/1986	Guenthner et al.	395/250
4,709,329	11/1987	Hecker	353/275
4,774,659	9/1988	Smith et al.	351/400
4,787,026	11/1988	Barnes et al.	395/700
4,827,403	5/1989	Dickson	355/250
4,860,201	8/1989	Smith et al.	395/800
4,933,845	6/1990	Hayes	395/325
4,949,252	8/1990	Hinge	371/29.1
5,003,471	3/1991	Gibson	353/250
5,029,074	7/1991	Medina et al.	353/325
5,091,846	2/1992	Sachs et al.	355/250
5,125,084	6/1992	Bagan et al.	395/375
5,168,568	12/1992	Thayer et al.	354/725
5,191,657	3/1993	Ludwig et al.	395/325
5,237,676	4/1993	Arimilli et al.	395/550
5,274,780	12/1993	Nakao	395/325

OTHER PUBLICATIONS

Digital Bus Handbook, Joseph DiGiacomo, ed., Chapters 5–7, McGraw-Hill Publishing Company, 1990.

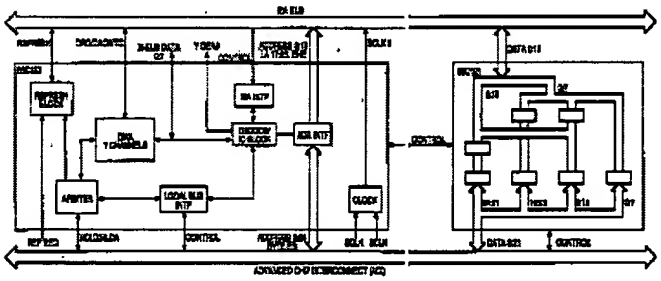
Primary Examiner—Glenn A. Auve

Attorney, Agent, or Firm—Skjerven, Morrill, MacPherson, Franklin & Friel; Norman R. Klivans

[57] **ABSTRACT**

An open high-speed local system bus for a microcomputer system which is decoupled from I/O and provides a consistent interface to the CPU subsystem, memory subsystem, graphics subsystem and peripheral subsystem. The local system bus supports discrete and burst transactions, pipelining in both the transactions, multiple microprocessor and distributed interrupts.

44 Claims, 92 Drawing Sheets



L5: Entry 1 of 1 File: USPT Oct 19, 1999

DOCUMENT-IDENTIFIER: US 5970069 A
TITLE: Single chip remote access processor

Abstract Text (1):
A single chip integrated remote access processor circuit has a plurality of communication interface units, including a local area network (LAN) interface unit, a first multi-protocol serial wide area network (SWAN) interface unit, a telephony coder-decoder interface unit and a peripheral component interface (PCI) unit. A data routing control circuit is coupled to the plurality of communication interface units for controlling data transfer between the interface units.

Brief Summary Text (10):
The single chip integrated remote access processor of the present invention has a plurality of communication interface units, including a local area network (LAN) interface unit, a first multi-protocol serial wide area network (SWAN) interface unit, a telephony coder-decoder interface unit and a peripheral component interface (PCI) unit. A data routing control circuit is coupled to the plurality of communication interface units for controlling data transfer between the interface units.

Detailed Description Text (3):
FIG. 2a is a block diagram illustrating a particular application of the remote access processor of the present invention which provides diverse connectivity to a network server 30 at a corporate headquarter or ISP server site. Remote access processor 34 includes local area network (LAN) interface 36, peripheral component interface (PCI) 38, and a plurality of serial wide area network (SWAN) interfaces 40, 42 and 44. Remote access processor 34 is preferably implemented on a single semiconductor integrated circuit within network server 30. PCI interface 38 allows communication between remote access processor 34 and the host processor within network server 30.

Detailed Description Text (8):
FIG. 3 is a block diagram of remote access processor 34. Remote access processor 34 is implemented on a single integrated circuit chip having a plurality of inputs and outputs. In one embodiment, the integrated circuit chip is manufactured with a 3.3 volt, 0.35 micron CMOS fabrication technology and is packaged within a 256 position plastic ball grid array having nominal dimensions of 27 mm.times.27 mm.times.2.1 mm. The communication interfaces include LAN interface 36, PCI interface 38 and SWAN interfaces 40, 42a, 42b and 44. SWAN interface 40 is a multi-protocol SWAN interface. SWAN interfaces 42a and 42b are time division multiplexer (TDM) serial interfaces for supporting two ISDN-BRI networks. SWAN interface 44 is a V.34 coder-decoder (CODEC) interface for coupling to a V.34 CODEC modem. The remaining inputs and outputs of remote access processor 34 include serial peripheral interface (SPI) 55, reference clock input 56, real time clock (RTC) input 57, battery back-up input 58, UART compatible serial I/O port 60, 40 Mhz clock input 62, boundary scan test I/O 64, 32-bit local memory interface 66 and general purpose interrupt/control I/O 70.

Detailed Description Text (9):
V.34 CODEC interface 44 is coupled to V.34 interface controller 72, which provides a serial digital interface to an external V.34 CODEC and digital-to-analog converter. V.34 CODEC interface 44 can also be routed to an external ISDN-BRI channel.

Detailed Description Text (10):
Ethernet controller 74 is coupled to Ethernet interface 36. Ethernet controller 74 is based on the E-110 Core Logic, which is available from LSI Logic Corporation, and provides a connection to an external Media Independent Interface (MII). Multi-protocol SWAN controllers 76a, 76b, 76c and 76d support four multi-protocol WAN interfaces. SWAN controller 76a is coupled directly to multi-protocol serial WAN interface 40 and is coupled to TDM serial interfaces 42a and 42b through 4-to-1 time slot multiplexers 78a and 78b. SWAN controller 76a can support a Frame Relay protocol over a leased line that is coupled to Multi-protocol WAN interface 40, for example. SWAN controllers 76b, 76c and 76d are coupled to TDM serial interfaces 42a and 42b through time slot multiplexers 78a and 78b. Time slot multiplexers 78a and 78b provide independent transmit and receive TDM interfaces at data rates of up to 8 Mhz, for example. These interfaces support ISDN, IOM2, IDL, Mitel ST-Bus, AT&T P7270 and T1/E1 communication applications, for example. Four-bit serial controller 80 is coupled to SPI interface 55 for providing synchronous, a bidirectional serial peripheral interface control for external ISDN transceivers coupled to TDM serial interfaces 42a and 42b.

Detailed Description Text (22):
If the data packet is being transferred through V.34 interface controller 72, CPU 90 executes a V.34 modem algorithm (e.g. a data pump algorithm) which performs the digital signal processing or "modulation" portion of the conversion between the digital data packets and the analog signals that are transmitted over the public switched telephone network. The modulated digital signals are then provided from CPU 90 to V.34 interface controller 72 for serial transfer to an external V.34 Codec and digital-to-analog converter which is coupled to interface 44. The V.34 Codec performs the final digital-to-analog conversion. Multiply and accumulator circuit 96 supports the digital signal processing performed by CPU 90 with the V.34 modem algorithm.

Detailed Description Text (23):
For ISDN-BRI interfaces, voice, data and facsimile information are integrated over a single serial output. This is accomplished by transferring the corresponding data packets through separate SWAN controllers, such as SWANs 76b, 76c and 76d and then time-multiplexing the data packets together through one of the time slot multiplexers 78a and 78b.

Detailed Description Text (25):
When a data packet is received at one of the serial WAN interfaces 40, 42a or 42b or at LAN interface 36, the data packet is passed to DMA controller 82 which temporarily stores the data packets in local memory. CPU 90 maintains at least one, and preferably two, "receive" linked lists in memory 84 for each channel of DMA controller 82. The receive linked list includes a BMD for each received data packet that is stored in local memory and corresponds to that DMA channel.

Detailed Description Text (28):
When serial data is received at V.34 Codec interface 44, the data represents a sampling of the analog signal transmitted over a public switched telephone network. The sampling is performed by an external V.34 Codec (not shown) coupled to V.34 Codec interface 44. As in the case where serial data is received at one of the LAN or WAN ports DMA controller 82 collects the serial data and stores the data in local memory, through memory controller 110. Once the data samples have been stored in local memory, CPU 90 retrieves the data samples and demodulates the samples into digital data packets by performing a digital filter and data pump function through

the V.34 digital signal processing algorithm. CPU 90 then stores the data packets back into local memory through memory controller 110, and notifies the routing software executed by CPU 90 to look for the packets in local memory. The routing software then retrieves the packets from local memory and determines their destination, as discussed above.

Detailed Description Text (32):
TDM serial interface port 42a is coupled to ISDN transceiver 168. ISDN front end circuit 170 is coupled between ISDN transceiver 168 and ISDN network 172. Similarly, multi-protocol SWAN interface 40 is coupled to framer 166. A DS-1 CSU/E1 isolation transformer circuit 176 is coupled between framer 166 and SWAN network 178. Flash ROM 164 provides 8-bits of instruction data over local memory bus 160. Remote access processor 34 controls the operation of ISDN transceiver 168 and framer 166 using 8-bit bus 174. In an alternative embodiment, in which a Motorola ISDN transceiver is used, SPI interface 55 provides control for the transceiver.

Detailed Description Text (33):
V.34 Codec interface 44 is coupled to V.34 Codec 180. In one embodiment, V.34 Codec 180 is a SGS/Thompson 7546 V.34 Codec. V.34 DAA interface front end circuit 182 is couple between V.34 Codec 180 and public switched telephone network 184. Circuits 180 and 182 receive the serial V.34 processed digital signals from remote access processor 34 and perform the final digital-to-analog conversion for sending the data over network 184.

Detailed Description Text (83):
3.2.4 UART Serial Interface 60

Detailed Description Text (84):
The Serial UART Interface provides a serial port for connecting to a console and allows for RAP system monitor and debug.

Detailed Description Text (103):
3.2.6 TDM Serial Interface 42a and 42b

Detailed Description Text (234):
DMA Controller 82 is illustrated in FIG. 23. DMAC 82 moves data between the serial communications controllers (SWANs 76a-76d, E110 ethernet controller 74 and V.34 codec interface 72) and either the integrated SRAM 84 or Local Memory Interface 66. The DMAC buffers data to and from the byte-wide communications controllers and bursts 1,2,4 or 8 words to the Local Memory Interface 66. Arbitrary byte alignment is supported.

Detailed Description Text (237):
Data frames are transferred between the serial interface controllers and the 32-bit Local Memory or internal SRAM 84. Both interfaces are maintained and monitored by DMAC 82. An additional End Of Frame (EOF) bit is transferred with each byte through each queue 250 and 252. The EOF bit is active (one), one byte after the last data byte of a frame.

Detailed Description Text (244):
In the receive direction, each channel of DMAC 82 has an associated byte wide Receive Data Queue within circuit 252. In additicn to the 8 bits of data, the Receive Data Queues have 1 bit indicating the End of Frame (EOF). The EOF bit is set in the Receive Data Queue byte location immediately following the last data byte of a frame. Bits 0-7 of the byte associated with the EOF bit are used as error flags from the serial interface controller.

Detailed Description Text (266):
The V.34 Modem in RAP 34 includes CW4011 resident software, a 16-bit multiply/accumulate 96 (shown in FIG. 3) and the V.34 Codec Interface 44. The V.34

Codec Interface 44 provides a serial interface to an external V.34 Codec.

Detailed Description Text (268):
The V.34 interface block 72 is the interface between RAP 34 and the off-chip line and voice codec/DAA combination, as shown in FIG. 39. The design of the V.34 interface block is optimized for the LSI Logic codec/DAA reference design, but other implementations are possible. The V.34 interface includes a serial Line Data path 280 (this is the path for the digital samples coming from and going to the telephone line), a serial Voice Data path 282 (this is the path for digital samples coming from and going to the microphone and speaker), and a set of control and status signals. The serial Line Data path includes FIFOs 284a and 284b and serial interface 286, which are located on RAP 34 and communicate with DMAC 82 (shown in FIG. 3), and external Line Codec 288. The serial Voice Data path includes FIFOs 290a and 290b, serial interface 292, and external Voice Codec 294. The Line Data interface and Voice Data interface are serial connections to external Codecs 288 and 294. The interfaces are based on Time Division Multiplexed protocols specific to the codecs used. V.34 Codec Interface Block 72 further includes an modem control register 296 which is coupled to an external DAA/Line control circuit 298 that is associated with Codecs 288 and 294.

Detailed Description Text (305):
FIG. 61 is a block diagram which illustrates WAN port configurability. Two Time Slot Multiplexers (TSMs) 340 and 342 provide TDM highways for ISDN BRI, channelized T1/Eland other TDM applications. The four SWAN controllers 76a-76d and two Serial Peripheral Interfaces (SPI) 80 connect to both TSMs. Serial Channel to time slot mapping is programmed in the Switch Table RAM associated with each TSM. Alternately, SPI #1 may be mapped to external signal pins as a general purpose serial port.

CLAIMS:

1. A single chip remote access processor for receiving and transmitting data packets having headers with destination addresses, the processor comprising:

a plurality of communication interface units adapted to transmit and receive the data packets and comprising a local area network (LAN) interface unit, a first multi-protocol serial wide area network (SWAN) interface unit, a telephony coder-decoder interface unit and a peripheral component interface (PCI) unit;

a multi-channel direct memory access (DMA) controller which is coupled to the plurality of communication interface units and comprises a plurality of channels and a contents addressable memory (CAM), wherein each channel comprises a transmit queue and a receive queue and wherein the CAM comprises a contents compare input coupled to the receive queues so as to receive the destination address of each received data packet and comprises a compare output that identifies which of a plurality of receive linked lists each data packet received by the receive queues is to be appended based on a comparison of the respective destination address;

a central processing unit (CPU) is adapted to maintain a transmit linked list of transmit buffer memory descriptors (BMDs) for each channel of the DMA controller and to maintain the plurality of receive linked lists of receive BMDS, wherein each transmit BMD corresponds to a data packet to be transmitted through the respective channel and wherein each receive BMD corresponds to a data packet received through one of the channels.

11. The single chip remote access processor of claim 10 wherein:

the telephony coder-decoder interface unit comprises a serial interface for receiving modulated digital samples; and

the CPU comprises means for demodulating the digital samples, translating the digital samples into data packets, and routing the data packets to a selected one of the communication interface units.

14. A remote access processor comprising:

an internal transfer bus;

a plurality of communication interface units comprising a local area network (LAN) interface unit, a first multi-protocol serial wide area network (SWAN) interface unit, a telephony coder-decoder interface unit and a peripheral component interface (PCI) unit, wherein the telephony coder-decoder interface unit comprises a serial interface for receiving modulated digital samples;

a multi-channel direct memory access (DMA) controller coupled to the internal transfer bus and having a first channel coupled to the LAN interface unit, a second channel coupled to the multi-protocol SWAN interface unit and a third channel coupled to the telephony coder-decoder interface unit;

a central processing unit (CPU) coupled to the internal transfer bus which is adapted to receive a first data packet through the first or second channel of the DMA controller, modulate the first data packet for transmission through the telephony coder-decoder interface unit and route the modulated first data packet to the telephony coder-decoder interface unit, through the third channel of the DMA controller and is adapted to receive the modulated digital samples from the telephony coder-decoder through the third channel of the DMA controller, demodulate the digital samples, translate the demodulated digital samples into a second data packet, and route the second data packet to the LAN interface unit or the multi-protocol SWAN interface unit, through the first and second channels, respectively, of the DMA controller;

a memory coupled to the DMA controller and the CPU; and

wherein the PCI interface unit is coupled to the internal transfer bus.

Previous Doc Next Doc Go to Doc#

FOREIGN PATENT DOCUMENTS

FOREIGN-PAT-NO	PUBN-DATE	COUNTRY	US-CL
WO 02/073420	September 2001	WO	

OTHER PUBLICATIONS

Anderson, Don and Tom Shanley, PCI System Architecture, Fourth Edition, 1999, pp. I-xlii, 1-6, 143-151, 351-409, Addison-Wesley, Boston.
APT Technologies, Inc., "Serial ATA: High Speed Serialized AT Attachment", Revision 1.0, Aug. 29, 2001, pp. 1-7, 23-36.
Hosler, Brad, "Programming Interface for Bus Master IDE Controller", [online], Revision 1.0, May 16, 1994, pp. 1-6. Retreived from the Internet at <URL: http://www.bswd.com/idems100.pdf>.
"PCI IDE Controller Specification", [Online], Revision1.0, Mar. 4, 1994, pp. 1-5. Retrieved from the Internet at <htt://www.bswd.com/pciide.pdf>.
PCI Special Interest Group, "PCI Local Bus Specification", PCI Local Bus, Revision 2.2, Dec. 18, 1998, pp. i-xx, 1-6, & 189-220.
PCT International Search Report, Dec. 15, 2003, for International Application No. PCT/US 03/22943.
PCT International Search Report, Dec. 17, 2003, for International Application No. PCT/US 03/22784.
U.S. patent application Ser. No. 10/205,664, filed Jul. 24, 2002, entitled "Method, System, and Program for Controlling Multiple Storage Devices", invented by S. Bissessur & D. R. Smith.

ART-UNIT: 2182

PRIMARY-EXAMINER: Fleming; Fritz M.

ATTY-AGENT-FIRM: Dutta; Rabindranath Konrad Raynes & Victor LLP

ABSTRACT:

Provided are a method, system, and program for a local bus system. A memory address space in configured to control an I/O device. The memory address space is associated with a port coupled to the local bus system.

44 Claims, 6 Drawing figures

Previous Doc Next Doc Go to Doc#

h e b b g eeef c e e ge